Slide 0

Online Smoothing of Live, Variable-Bit-Rate Video

Jennifer Rexford[1], Subhabrata Sen[2], Jayanta Dey[2], Wu-chi Feng[3], James Kurose[2], John Stankovic[4], and Don Towsley[2]

[1] Network Mathematics
AT&T Labs – Research
Murray Hill, NJ 07974

[2] Dept. of Computer Science
University of Massachusetts
Amherst, MA 01003

[3] Comp. & Info. Sciences
Ohio State University
Columbus, OH 43210

[4] Dept. of Computer Science
University of Virginia
Charlottesville, VA 22903

Slide 1

Outline

- Live video with relaxed delay requirements
- Smoothing interactive and pre-recorded video
- Online sliding-window smoothing algorithm
- Performance evaluation with MPEG video traces
- Conclusions and ongoing work

Slide 2

Efficient Distribution of Live Video

Emerging video applications

- Distance learning, broadcasts of course lectures
- MBone, video teleconferencing
- Television news, concerts, sporting events

→ many clients participating without transmitting

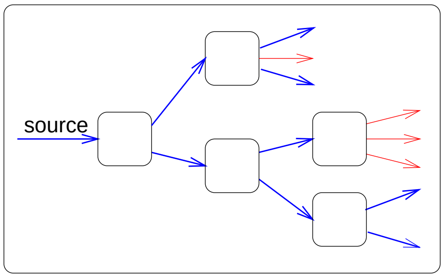
Networking challenges

- High bandwidth requirements of compressed video
- Burstiness of compressed video on several time scales
- Clients and links with different bandwidth resources

→ mechanisms for variegated QoS & reducing burstiness

Slide 3

Online Smoothing of Live Video



Online smoothing at source or intermediate server

- Can tolerate larger delay than interactive video
- Have only some video frames in advance
- Have only partial knowledge of frame sizes

→ hybrid of interactive & pre-recorded video smoothing

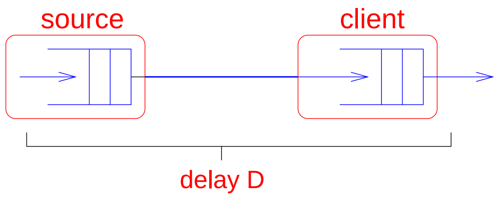
Slide 4

Smoothing Interactive Video

For interactive applications, such as teleconferencing:

- Must tightly control end-to-end delay
- Do not have frames in advance
- Can only estimate future frame sizes

→ modest smoothing by averaging frame sizes at source



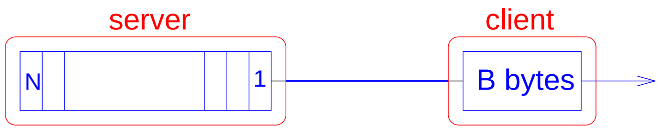
Slide 5

Smoothing Pre-Recorded Video

For pre-recorded applications, such as video-on-demand:

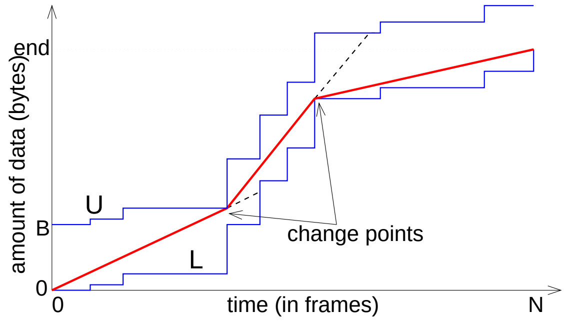
- Want to limit bandwidth requirements
- Have all video frames in advance
- Have full knowledge of frame sizes

→ aggressive prefetching into client playback buffer



Slide 6

Smoothing Pre-Recorded Video



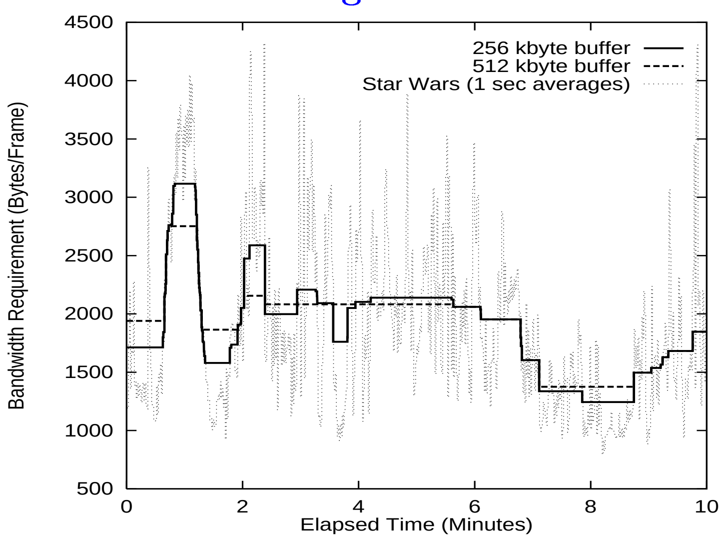
Given frame sizes $\ell_1, \ldots, \ell_N$ and a B-byte buffer,

- Lower constraint: $L_k = \sum_{i=0}^{k} \ell_k$
- Upper constraint: $U_k = L_k + B$

$\rightarrow O(N)$ algorithm minimizes peak and variation of rates

Slide 7

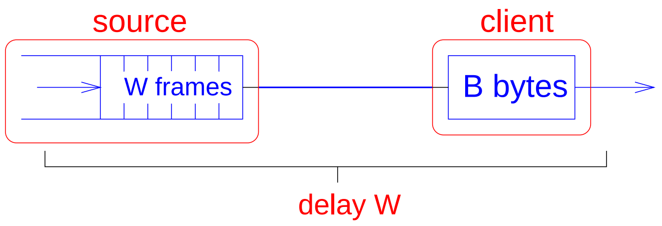
Smoothing *Star Wars*



Slide 8

Smoothing of Non-Interactive Live Video

Source can delay the live feed by W time units:



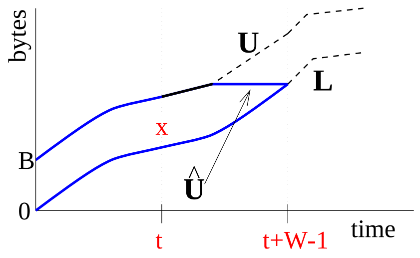
Larger W reduces bandwidth requirement but requires

- Larger buffer at the source (store W frames)
- Larger processing load (compute over W frames)
- Larger playback delay (at least W time units)

Slide 9

Online Smoothing Constraints

Source has W frames ahead of current time t:



Conservative view of the prefetching constraints:

$$\hat{U}_k^t = \min\left(L_k + B, \sum_{i=1}^{t+W-1} \ell_i\right)$$

Slide 10

Sliding-Window Smoothing

Apply the offline smoothing algorithm with

- Modified constraint curves L and $\hat{U}$
- Client buffer occupancy at time t
- Video completion at end of W frames

$\rightarrow$ $O(W)$ online computation at time t

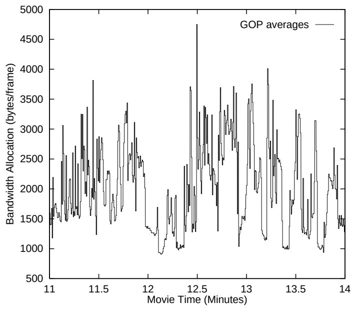
Reduce complexity at the source

- Execute online algorithm every α time units
- Limit computation to schedule next α steps
- Reduce the number of rate changes

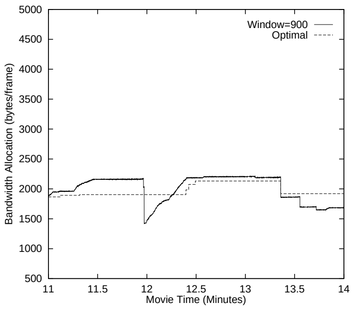
$\rightarrow$ at most $O(W)$ work every α time units

Slide 11

Smoothing Example: *Star Wars*



12-frame GOP averages



$W = 30$ sec, $B = 0.5$ MB, $\alpha = 1$

$\rightarrow$ online smoothing close to optimal offline schedule

Slide 12

Performance Evaluation

Algorithm parameters:

- Window size W
- Client buffer size B
- Slide length α

Performance metrics:

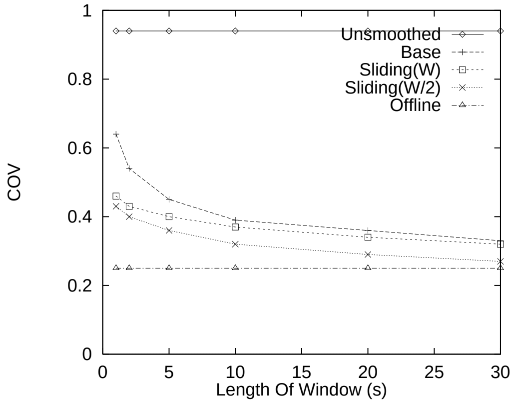
- Coefficient of variation (standard-deviation/mean)
- Peak rate (worst-case bandwidth requirement)
- Effective bandwidth (given buffer and loss rate)

for MPEG-1 encodings of *MTV*

Slide 13

Coefficient of Variation vs. Window Size W

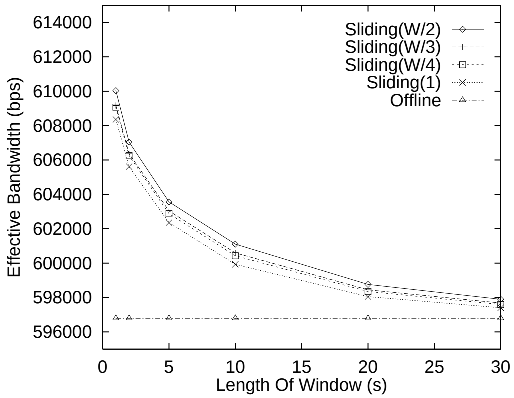
With prefetch buffer $B = 512$ kilobytes:



- Smoothing significantly reduces variability
- Benefit of placing I frame at end of window
- Online algorithms approach offline algorithm
- Ten-second window achieves main benefit

Effective Bandwidth vs. Slide Length α

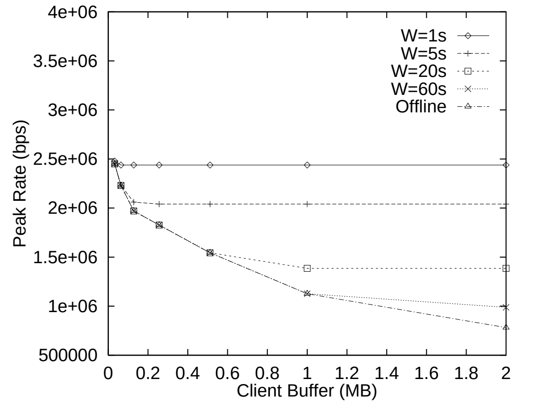
With a 512-kilobyte prefetch buffer, a 64-kilobyte switch buffer, and a loss probability of 0.001:



- Unsmoothed video requires 740 kbits/sec
- Smoothing approaches mean of 590 kbits/sec
- Online algorithm not sensitive to slide length

Peak Rate vs. Buffer Size B

With slide length $\alpha = W/2$:



- Smoothing limited by either window or buffer
- Larger buffer sizes offer diminishing returns
- Window size should scale with buffer size

Slide 16

Conclusions

- Motivation for smoothing non-interactive live video
- Online algorithm based on prefetch constraints
- Control over computational load with slide length

Ongoing work

- Heuristics for predicting future frame sizes
- Prefetching based on buffer constraints at the source
- Integration with packet retransmission protocols